ATTACHMENT 2

**UNITED STATES DISTRICT COURT
NORTHERN DISTRICT OF CALIFORNIA
SAN JOSE DIVISION**

CISCO SYSTEMS, INC.,

       Plaintiff,

   v.

ARISTA NETWORKS, INC.,
        Defendant.

Case No. 5:14-cv-05344-BLF (PSG)

**OPENING EXPERT REPORT OF KEVIN ALMEROTH
REGARDING COPYING**

SUBMITTED ON BEHALF OF CISCO SYSTEMS, INC.

42.   At the heart of each Cisco router and switch is Cisco's proprietary operating system, called the Cisco Internetworking Operating System ("IOS"). IOS is a complex computer program that controls a router/switch. According to Cisco, IOS is one of its "core technolog[ies]"[1] and that it is one of the company's most valuable assets—a "critical component of Cisco's business."[2]

43.   Originally developed in the 1980s, IOS has undergone continual enhancement over the years, and I understand Cisco has invested hundreds of millions of dollars developing IOS.[3] Since the introduction of IOS in the mid-1980s, Cisco continuously has upgraded, tested, and improved IOS through multiple releases. For example, IOS today operates on numerous Cisco routers as well as many of Cisco's other network products such as switches and gateways.[4]

44.   As a software architecture, IOS provides the unifying principles around which an internetwork can be maintained cost-effectively over time. It can be upgraded to adapt to changing technologies (hardware and software) as they evolve and are improved. IOS can be thought of as an internetworking brain, a highly intelligent administrative unit that manages and controls complex, distributed network resources, and functions.

45.   Since its release in the mid-1980s, IOS has undergone continual enhancement over the years, and Cisco developed specific and novel solutions in a number of protocol areas. These solutions are embodied in at least its programs.

46.   Cisco protects its IOS source code and does not disclose it publicly.

---

[1] CSI-CLEO3838924, Exhibit 2 to Cisco's Motion for Preliminary Injunction, *Cisco Systems v. Huawei* ("Cisco's PI Motion"), Declaration of Charles Giancarlo ("Giancarlo Decl."), ¶ 7.

[2] *Id.* at ¶12.

[3] *Id.*

[4] *See id.,* ¶¶ 3, 5.

47.   I have personally inspected portions of the IOS source code in support of preparing this report.

B.   The IOS CLI Computer Program

48.   In networking systems before IOS, complete configuration files were loaded on network devices using remote servers. In these older system, errors in the configuration files often would not be noticed until the router was activated. What this meant was that a new configuration file was needed to correct any error that may have been included in an initial configuration file. This process was a slow and inefficient way of configuring network devices.

49.   In the mid-1980s, Cisco endeavored to solve this problem.[5] Cisco's engineering team—lead by Kirk Lougheed—came up with the idea to create a unique CLI computer program.[6] This decision was an important one. As is the case with personal computers and workstations, the user interface computer program for a network device is a critical component of the system—it is the portal by which the user interacts with the device. It gives the operating system its distinct character, and distinguishes it from other operating systems.

50.   The computer program that Mr. Lougheed and his fellow Cisco engineers chose to build was one that would enable configuration changes directly on a Cisco router.[7] A CLI computer program allows a user to enter a "command" into a text-based input system in order to provide information (*e.g.,* a status request or a configuration command) to, and receive

---

[5]   *See generally* Deposition transcripts of Kirk Lougheed; Conversation with Kirk Lougheed (June 3, 2016).

[6]   *See generally* Deposition transcripts of Kirk Lougheed; Conversation with Kirk Lougheed (June 3, 2016).

[7]   Conversation with Kirk Lougheed (June 3, 2016); *see generally* Deposition Testimony of Kirk Lougheed; Abhay Roy; Adam Sweeney; Anthony Li; Devadas Patil; Greg Satz; Hugh Holbrook; Phillip Remaker; Ramanthan Kavasseri; and Tong Liu; *see also infra* Section V(C) (discussing creativity and originality).

information from, the device. When a user enters a command, the device typically provides some type of feedback to the user, for example, command confirmation or the output of executing such a command.

51.   In choosing to develop a CLI computer program (as opposed to other alternatives, *e.g.*, a graphical user interface), the Cisco engineers faced endless aesthetic choices for each of the numerous commands now found in the Cisco IOS CLI computer program and to select an elaborate structure and organization for these commands.[8] Neither the commands nor the structure and organization of the commands were dictated by technical requirements—they could have comprised different letters or numbers and have been organized in various different number of ways.[9] Additionally, Cisco designed expressive textual outputs that are used by the CLI computer program when providing feedback to the user and created an extensive set of command definitions as part of the program's help system. Like the initial choice of a CLI computer program, each of these subsequent steps in the development of the program was guided by the creativity and personal preferences of Cisco's engineers.[10]

52.   The Cisco IOS CLI is the product of decades of investment and creative endeavor by Cisco. Cisco also has spent years developing comprehensive user documentation and user

---

[8] Conversation with Kirk Lougheed (June 3, 2016); *see generally* Deposition Testimony of Kirk Lougheed; Abhay Roy; Adam Sweeney; Anthony Li; Devadas Patil; Greg Satz; Hugh Holbrook; Phillip Remaker; Ramanthan Kavasseri; and Tong Liu; *see also infra* Section V(C) (discussing creativity and originality).

[9] Conversation with Kirk Lougheed (June 3, 2016); *see generally* Deposition Testimony of Kirk Lougheed; Abhay Roy; Adam Sweeney; Anthony Li; Devadas Patil; Greg Satz; Hugh Holbrook; Phillip Remaker; Ramanthan Kavasseri; and Tong Liu; *see also infra* Section V(C) (discussing creativity and originality).

[10] *See generally* Deposition transcripts of Kirk Lougheed; Conversation with Kirk Lougheed (June 3, 2016); *see generally* Deposition Testimony of Kirk Lougheed; Abhay Roy; Adam Sweeney; Anthony Li; Devadas Patil; Greg Satz; Hugh Holbrook; Phillip Remaker; Ramanthan Kavasseri; and Tong Liu; *see also infra* Section V(C) (discussing creativity and originality).

manuals to further educate and train customers on the use of IOS and its CLI.[11] As a result of

this hard work and investment, I understand Cisco believes that its CLI also is a valuable asset

and critical component of its business.[12]

C.   IOS's Creative CLI Commands & Hierarchies

53.   As part of its CLI development, Cisco developed a distinctive and elegant syntax

and structure for the commands that are used by the IOS CLI. As the IOS documentation

explains:[13]

Understanding Command Syntax

Command syntax is the format in which a command should be entered in the CLI. Commands include the name of the command, keywords, and arguments. Keywords are alphanumeric strings that are used literally. Arguments are placeholders for values that a user must supply. Keywords and arguments may be required or optional.

Specific conventions convey information about syntax and command elements. Table 5 describes these conventions.

---

[11]   CSI-CLEO3838924, Giancarlo Decl., ¶ 12.

[12]   CSI-CLEO3838924, Giancarlo Decl., ¶ 12.

[13]   CSI-CLI-00226710 at CSI-CLI-00226747-48.

Table 5   **CLI Syntax Conventions**

| Symbol/Text	Function	Notes
< > (angle brackets)	Indicate that the option is an argument.	Sometimes arguments are displayed without angle brackets.
A.B.C.D.	Indicates that you must enter a dotted decimal IP address.	Angle brackets (< >) are not always used to indicate that an IP address is an argument.
WORD (all capital letters)	Indicates that you must enter one word.	Angle brackets (< >) are not always used to indicate that a WORD is an argument.
LINE (all capital letters)	Indicates that you must enter more than one word.	Angle brackets (< >) are not always used to indicate that a LINE is an argument.
<cr> (carriage return)	Indicates the end of the list of available keywords and arguments, and also indicates when keywords and arguments are optional. When <cr> is the only option, you have reached the end of the branch or the end of the command if the command has only one branch.	—

54.   Cisco's command expressions are organized hierarchically into groups and sub-groups of command expressions (as opposed to, for instance, having no organizational structure). For example, I understand Cisco has asserted various command hierarchies in this case including, *e.g.*, the "aaa" command hierarchy, "bgp" command hierarchy, "clear" command hierarchy, "dot1x" command hierarchy, "ip" command hierarchy, "ipv6" command hierarchy, "neighbor" command hierarchy, "show" command hierarchy, "snmp-server" command hierarchy, "spanning-tree" command hierarchy, "vrrp" command hierarchy, among other command expressions and hierarchies.

55.   Within a given command hierarchy, all of the commands start with the same word. For example, all of the commands within the "aaa" command hierarchy start with the "aaa." Additional sub-hierarchies within a command hierarchy that Cisco has asserted in this case include, *e.g.*, "ip dhcp" subhierarchy, "ip igmp" sub-hierarchy, "ip msdp" sub-hierarchy, "ip

ospf" sub-hierarchy, "ip pim" sub-hierarchy, "ipv6 nd" sub-hierarchy, "ipv6 ospf" sub-hierarchy, "show interfaces" subhierarchy, and "show ipv6" sub-hierarchy.

56.   Within a given command sub-hierarchy, all of the commands start with the same two words. For example, all of the commands within the "ip dhcp" sub-hierarchy start with "ip dhcp." And there can be further sub-hierarchies within a given sub-hierarchy. One way to visually illustrate the hierarchy and organization of Cisco's command expressions is through the use of a tree structure, for example:[14]
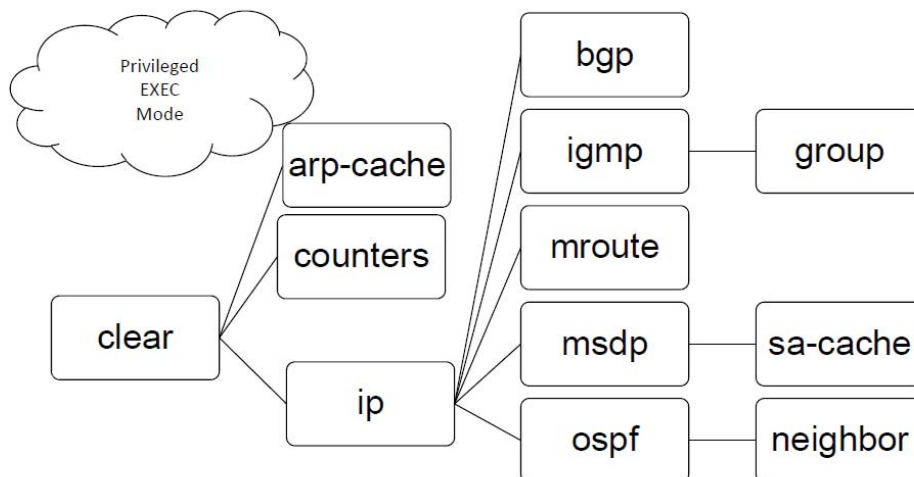
("clear" hierarchy in "Privileged EXEC" mode in IOS 11.0)

---

[14]   *See* Exhibit D to Cisco's Responses to Arista Interrogatory Nos. 2 and 16. The images below were taken from Cisco's interrogatory response, and I agree with Cisco's depiction of these hierarchies.
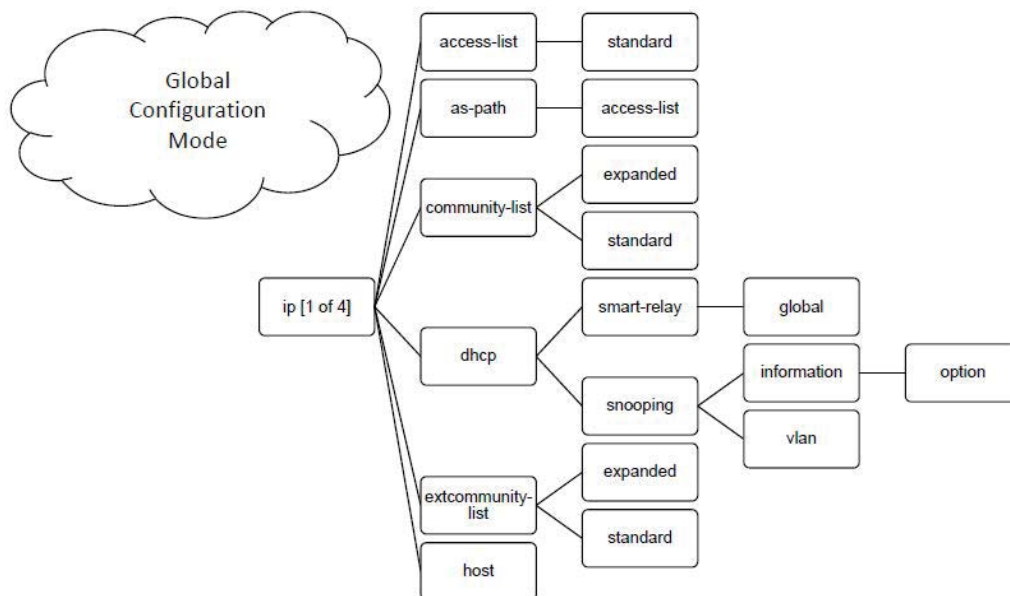
("ip" hierarchy in "Global Configuration" model in NX-OS 6.2)

57.   I understand this hierarchical arrangement of Cisco's CLI command expressions was originally created by Cisco engineer Kirk Lougheed in 1986.[15] That arrangement was first documented in the "Cisco Systems ASM/AGS User Manual and Configuration Guide Version 5.2."[16] As command expressions were added in subsequent versions of Cisco's copyrighted operating systems, the hierarchical arrangement of command expressions was modified and extended by the addition of more expressions.[17]

D.   Modes/Prompts

58.   In the IOS CLI, command "modes" are used to navigate the CLI and perform basic device startup, configuration, and monitoring tasks. "The CLI command mode structure is

---

[15]   Cisco's Third Supplemental Response to Arista's Interrogatory No. 16; Conversation with Kirk Lougheed (June 3, 2016); *see generally* Deposition Testimony of Kirk Lougheed.

[16]   CSI-CLI-00358622 to CSI-CLI-00358654.

[17]   Cisco's Third Supplemental Response to Arista's Interrogatory No. 16.

hierarchical, and each mode supports a specific set of commands."[18] Each mode also has an associated visual "prompt" that helps the user identify which mode they are in and, therefore, which commands are available.[19] Not unlike multi-word command expressions, creating the precise modes and their associated prompts associated with IOS was done through a creative process since professional judgment and subjective decisions by Cisco's engineers were required to create them. The IOS command modes and their associated prompts are discussed below.

59.   "User EXEC Mode" is the default command mode for the CLI. The EXEC commands available at the user EXEC level are a subset of those available at the "privileged EXEC" mode. In general, the user EXEC commands allow a user to connect to remote devices, change terminal settings on a temporary basis, perform basic tests, and list system information. The prompt for user EXEC mode is the name of the device followed by an angle bracket, e.g., Router>.

60.   Another example of an IOS mode is "Privileged EXEC Mode." Privileged EXEC mode is password protected, and allows the use of all EXEC mode commands available on the device. To enter privileged EXEC mode from user EXEC mode, a user will enter the "enable" command. The privileged EXEC mode prompt consists of the devices' host name followed by the pound sign, *e.g.*, Router#.

61.   A third type of IOS mode is "Global Configuration Mode." "Global Configuration Mode" is used for configuration commands that generally apply to features that affect a system as a whole, rather than just one protocol or program. Once a user is in Privileged EXEC mode they can access "Global Configuration Mode" through the use of the "enable"

---

[18]   CSI-CLI-00226710 at CSI-CLI-00226743.

[19]   CSI-CLI-00226710 at CSI-CLI-00226745.

command. The router prompt for global configuration mode is indicated by the term config in parenthesis: Router(config)#.

62.   IOS prompts help a user identify which mode they are in and, therefore, which commands are available for use. Each mode described above has the following corresponding prompts:[20]

Table 3   CLI Command Modes

Command Mode	Access Method	Prompt	Exit Method	Mode Usage
User EXEC	Log in.	Router>	Issue the logout or exit command.	• Change terminal settings. • Perform basic tests. • Display device status.
Privileged EXEC	From user EXEC mode, issue the enable command.	Router#	Issue the disable command or the exit command to return to user EXEC mode.	• Issue show and debug commands. • Copy images to the device. • Reload the device. • Manage device configuration files. • Manage device file systems.
Global configuration	From privileged EXEC mode, issue the configure terminal command.	Router(config)#	Issue the exit command or the end command to return to privileged EXEC mode.	Configure the device.
Interface configuration	From global configuration mode, issue the interface command.	Router(config-if)#	Issue the exit command to return to global configuration mode or the end command to return to privileged EXEC mode.	Configure individual interfaces.
Line configuration	From global configuration mode, issue the line vty or line console command.	Router(config-line)#	Issue the exit command to return to global configuration mode or the end command to return to privileged EXEC mode.	Configure individual terminal lines.

63.   The command modes and prompts were first documented in the "Cisco Systems ASM/AGS User Manual and Configuration Guide Version 5.2."[21] I understand that they were created by at least the date of that document: July 20, 1986.[22]

---

[20] CSI-CLI-00226710 at CSI-CLI-00226745.

E.   Screen Displays -- Responses/Outputs

64.   Another aspect of Cisco's CLI are the screen outputs/displays generated by the CLI in response to a command input. Cisco's CLI outputs contain context sensitive response information that is displayed to a user, such as configuration information of the device and networking information related to a device, among other command-specific outputs. The CLI outputs contain unique and varied information types in textual form and are organized in unique structural arrangements, all of which were created by Cisco. Exemplary CLI outputs for certain asserted Cisco CLI commands are provided below as examples:

```
Router# show ip route

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set
```

(output for "show ip route"[23])

```
Router# show interfaces atm 0/0/0
ATM0/0/0 is up, line protocol is up
 Hardware is cyBus ATM
 Internet address is 10.1.1.1/24
 MTU 4470 bytes, sub MTU 4470, BW 156250 Kbit, DLY 80 usec, rely 255/255, load 1/255
 Encapsulation ATM, loopback not set, keepalive set (10 sec)
 Encapsulation(s): AAL5, PVC mode
 256 TX buffers, 256 RX buffers,
 2048 maximum active VCs, 1024 VCs per VP, 1 current VCCs
 VC idle disconnect time: 300 seconds
 Last input never, output 00:00:05, output hang never
 Last clearing of "show interface" counters never
 Queueing strategy: fifo
 Output queue 0/40, 0 drops; input queue 0/75, 0 drops
 5 minute input rate 0 bits/sec, 1 packets/sec
 5 minute output rate 0 bits/sec, 1 packets/sec
     5 packets input, 560 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     5 packets output, 560 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

---

[21] CSI-CLI-00358622 to CSI-CLI-00358654.

[22] Cisco's Third Supplemental Response to Arista's Interrogatory No. 16.

[23] CSI-CLI-00408381, Cisco IOS IP Routing Protocols Command Reference, Release 12.4 (2005), at IP2R-553.

service providers. IOS-XR was created by engineers and uses a similar CLI to the traditional

IOS CLI discussed above.[26]

G.   IOS-XE

67.   IOS-XE is another Cisco operating system and is related to traditional IOS. IOS-

XE was built on a modular operating system and it provides scale and serviceability for both

enterprises and service providers. IOS-XE supports the complete set of Cisco IOS features

including the IOS CLI discussed above. IOS-XE also scales advanced service delivery without

impacting system performance, integrates applications in the network, improves security,

reliability, and simplicity, facilitates programmability for cloud service orchestration, supports

next-generation platform, among other features.[27]

H.   Cisco's NX-OS Platform

68.   NX-OS is Cisco's next generation operating system. NX-OS was created for

Cisco's Nexus platform and was built primarily for data center environments. Cisco NX-OS

provides a CLI as well as implementations of relevant networking standards and a variety of data

center related features. Cisco NX-OS runs on the Cisco Nexus Family of hardware-based

network switches, which include Cisco Nexus 7000, 5000, 4000, and 1000V Series Switches and

---

[26]   *See generally* http://www.cisco.com/c/en/us/products/ios-nx-os-software/ios-xr-software/index.html?CAMPAIGN=Cisc_3; *see also, e.g.*, Cisco copyrighted documentation submitted with the Copyright Office for this operating system as set forth in Cisco's responses to Interrogatory Nos. 24 and 25, which are incorporated here by reference.

[27]   *See generally* http://www.cisco.com/c/en/us/products/ios-nx-os-software/ios-xe/index.html; Cisco copyrighted documentation submitted with the Copyright Office for this operating system as set forth in Cisco's responses to Interrogatory Nos. 24 and 25, which are incorporated here by reference.

Cisco Nexus 2000 Series Fabric Extenders; Cisco MDS 9000 Family storage switches; and Cisco UCS 6100 Series Fabric Interconnects.[28]

IV.   ARISTA'S EOS PLATFORM

A.   EOS Overview

69.   Founded in 2004 by former Cisco engineers, Arista Networks ("Arista") is one of Cisco's competitors in the internetworking industry. According to Arista's CEO—a former Cisco executive—Cisco is viewed by Arista as a "fierce competitor."[29] Like Cisco, Arista sells switches with an operating system and command line interface computer program, referred to by Arista as the Extensible Operating System ("EOS"). According to Arista, its EOS "is the core of Arista cloud networking solutions for next-generation data centers and cloud networks."[30] The switches that Arista's sells with its EOS and CLI are based on 10GbE, 40GbE, and 100GbE platforms, and include at least its 7010T, 7280SE, 7150S, 7050TX, 7050SC, 7050OX, 7250OX, 7060CX, 7260X, 7300 series, and 7500R series switches.

70.   As the evidence below shows, Arista's purpose in creating EOS was to create a substitute for Cisco's IOS. EOS directly competes with IOS in the market such that if a competitor has an Arista switch running EOS they have no need for Cisco switches running IOS (or one of Cisco's other copyrighted operating systems).

---

[28]   *See* http://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/nx-os-software/data_sheet_c78-652063.pdf; Cisco copyrighted documentation submitted with the Copyright Office for this operating system as set forth in Cisco's responses to Interrogatory Nos. 24 and 25, which are incorporated here by reference.

[29]   CSI-CLl-00357842 at CSI-CLI-00357851.

[30]   *See* https://www.arista.com/en/products/eos.

71.   In support of my opinions, and during the process of preparing this report, I personally inspected and tested three different Arista switches running EOS, pictures of which I have provided along with this report. I have reproduced some images of the Arista switches I inspected and tested below as examples:

(Arista 7010T-48)

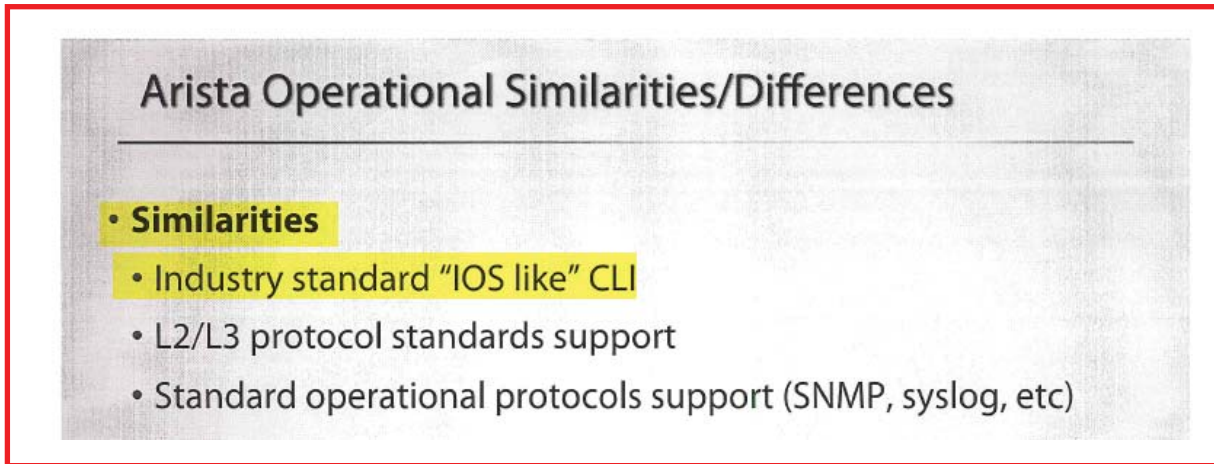(testing an Arista 7010T-48 running EOS)

(Arista DCS-7554)

(testing an Arista DCS-7554 running EOS)

B.   EOS CLI

72.   Like Cisco's IOS, Arista's EOS has a CLI that network engineers and other users can leverage to configure Arista switches. Arista promotes its EOS CLI as being similar to Cisco's IOS or "IOS like":[31]

---

[31]   Sadana Deposition, Exhibit 382, at 78.

73.   According to Arista, the EOS CLI is a "familiar industry standard CLI and runs a single binary software image across the Arista switching family."[32] Based on the evidence that I have reviewed in this case, Arista's use of the term "industry standard" is effectively a codename that Arista has used to refer to Cisco's IOS CLI. As described in more detail below, there is no CLI industry standard let alone an industry standard for Cisco's copyrighted works, which include elements such as screen displays, outputs, and prompts. Indeed, as one Arista employee wrote: "Juniper's JunOS is different enough from 'industry standard' (meaning: whatever Cisco did) . . . ."[33]

74.   Another Arista engineer, Lincoln Dale, gave a presentation at a QuestNet Conference in July 2013 where he told the audience that Arista's use of the term "industry standard" was Arista's internal way of referring to Cisco's IOS: "We say that our switches run an industry standard CLI. I guess that's my joke for saying it's the same as IOS." Arista's use of the term "industry standard" as a codename for Cisco's IOS CLI is further confirmed by Arista's "EOS CLI Conventions and Style Guidelines," which states that the "most important convention . . . is to following the industry standard," which defines that "industry standard" as

---

[32]   ANI-ITC-944_945-3473603.

[33]   ARISTANDCA1195413 (emphasis added).

29

HIGHLY CONFIDENTIAL – ATTORNEYS' EYES ONLY – SOURCE CODE

three Cisco operating systems (IOS, NX-OS, IOS-XR)—which are Arista's "preferred" models

to follow—and then a Juniper operating system (JunOS), which Arista lists as a fourth model.[34]

In other words, the EOS CLI is marketed as being familiar because it was modeled after Cisco's

IOS CLI, which was the industry leading CLI on the market at the time that Arista developed its

EOS CLI.

75.   As Arista's own executives and engineers also have explained to the public, the

EOS CLI was intentionally designed to be similar to Cisco's IOS CLI:

- "[A] Cisco CCIE expert would be able to use Arista right away, because we have a similar command-line interface and operational look and feel. Where we don't have to invent, we don't."[35]

- Arista tried to "[p]rovide familiar interfaces to ease adoption" including a "standard CLI that … retains familiar management commands" such that "80% [of Arista customers] tell [Arista] they appreciate the way they can leverage their deep [Cisco] IOS experience, as they can easily upgrade an aging [Cisco] Catalyst infrastructure to Arista."[36]

- "Familiar management interfaces, standard CLI … It's been very helpful for our customers to be able to rapidly adopt our products and integrate them into their environments … that our switches provide a familiar management interface so their existing tools and processes, screen scraping, automation, continue to work just as they did before."[37]

76.   Many other examples of Arista employees confirming that the EOS CLI was

designed to be similar to Cisco's IOS CLI are discussed below.[38]

---

[34]   ANI-ITC-944_94 0962624 at ANI-ITC-944_945-0962625.

[35]   CSI-ANI-00381280, John Gallant, "How Arista Networks Got Out In Front of the SDN Craze," Network World (Feb. 22, 2013).

[36]   Posting of Kenneth Duda to Arista EOS Central, "Linux as a Switch Operating System: Five Lessons Learned" (Nov. 5, 2013), *available at*https://eos.arista.com/linux-as-a-switch-operating-system-five-lessons-learned/.

[37]   Arista, *EOS Bits & Bytes - Episode 1 - Lessons Learned While Building a Network OS on Top of Linux*, Arista EOS Central - Video Library (Jan. 30, 2014), at 6:55–7:56, *available at* http://eos.arista.com/wpcontent/themes/aristaeos/video-lightbox.php?vid=ttp6lavHKGo.

[38]   *See infra* Section VI(A).

12 a few other protocols to make things look very
13 similar, we started prefacing our IP-only commands
14 with "IP." And that gave a very—what I thought
15 was a very elegant, symmetric, elegant way of
16 referring to different protocols within a
17 multi-protocol router.
18 So that is the history of the "IP address"
19 command.

104.   Mr. Lougheed provided similar testimony for specific multi-word command expressions as well such as "show ip route,"[51] "show spanning-tree,"[52] "IP routing,"[53] "show hosts,"[54] "clear" hierarchy,[55] and "timers basic RIP."[56]

105.   Another Cisco CLI command author, Mr. Abhay Roy, testified similarly. For instance, Mr. Roy testified that the creation of the command "bfd all-interfaces" was the result of looking at a variety of protocols, collectively discussing the best way to express the concept, considering how the command "fits into the bigger … pieces of organization of commands, what makes sense, [and] what is more aesthetically correct" within the framework of the system.[57] Mr. Roy also testified he considered many things when designing commands such as content, features, "what is being asked," and that during the creative process "you start with your best

---

[51]   Deposition Testimony of Kirk Lougheed Tr. at 331:6-23 (April 4, 2016).

[52]   *Id.* at 337:17-20.

[53]   Deposition Testimony of Kirk Lougheed Tr. at 145:3-25 (Nov. 20, 2015).

[54]   *Id.* at 168:21-169:16 (testifying that there were other command word options he could have chosen including "computers," "names," "systems," "network systems," "end systems").

[55]   *Id.* at 174:5-175:4 ("it seemed aesthetically pleasing to me").

[56]   *Id.* at 185:13-186:5.

[57]   Roy Deposition Tr. at 24:12-25; 26:2-9 (discussing that command creation involves considering "overall architecture purity"); 45:6-20 (testifying that when creating commands Cisco wanted to make "smart choices" that made sense from an "aesthetic perspective" and from "the alignment and architectural perspective").

guess," which "may or may not be the best which will eventually have the light of day, but you

go with your knowledge and your judgment."[58]

106. Another Cisco CLI command author, Mr. Devadas Patil, testified that the

command creation process is subjective and implicates various considerations that are open to an

author's own professional judgment:

- "Well, there is—the—the product owner, which is me, lead developer for the product, comes up with initial proposal, and it is, essentially, reviewed by a group of people that are highly experienced for—for usability and extensibility, and so on, so there are certain criteria that they look—look at, including usability, extensibility, aesthetics, etc."[59]

- "So there's a—there's a—there's a balance between future-proofing and—and verbosity, and—and the more you try to feature proof, the more verbose you can become, so it's more of a subjective column how you design, keeping all of these in mind, yeah."[60]

- "Yeah, so intuitiveness, extensibility, usability, aesthetics are all factors that we considered."[61]

107. Cisco engineer and CLI author Phillip Remaker's testimony confirms the same.

Mr. Remaker testified that commands, *e.g.*, "show inventory," were created at Cisco through a

collective discussion with other engineers (sometimes referred to as the Cisco "Parser Police")

during which many different word choices were considered:[62]

> 2 Q. In your view, what's creative about the
> 3 command "show inventory"? Strike that.
> 4 What is creative about the command "show
> 5 inventory"?
> 6 MR. NEUKOM: Objection. Calls for a legal
> 7 conclusion and personal opinion. Also off topic.
> 8 THE WITNESS: For this particular command,
> 9 we spent a lot of time in discussion and considered

---

[58]   *Id.* at 47:8-18.

[59]   Patil Deposition Tr. at 161:19-162:1 (Feb. 21, 2016).

[60]   *Id.* at 186:7-11.

[61]   *Id.* at 187:1-9.

[62]   Remaker Deposition Tr. at 114:2-15 (Mar. 31, 2016).

10 a lot of different ideas for how to go about doing
11 this command. And because we engaged a number of
12 people and spent a lot of serious time thinking
13 about the problem and how the customer would
14 interact with the command, I think that careful
15 consideration could be classified as creativity.

108.   Communications from other Cisco engineers further confirm that the process of

command expression creation is a subjective, creative endeavor. For example:

- Adam Sweeney (formerly of Cisco, now with Arista): "I agree with CLI naming is very subjective. . . . Review in this list gives us a chance to work towards consistency within this very subjective space."[63]

- Scott Lennartz (Cisco): "It is my belief that any exercise in naming is highly subjective, and there is rarely a 'right' answer . . . ."[64]

109.   This collaborative, creative, expressive process is what ultimately led to the Cisco

command syntax of Cisco's IOS CLI and "an aesthetic of the organization of the commands,"

which includes the "hierarchical notions, the modality, the organization of the commands, and

the choices of the words."[65] And, as Mr. Remaker testified, one of the reasons why Cisco chose

to organize commands into hierarchies was to "improve[] the readability of configurations."[66] In

other words: "Instead of having a single configuration line with a lot of attributes, it makes more

sense to have individual lines expressing each individual attribute."[67]

---

[63]   CSI-CLI00608716.

[64]   CSI-CLI00608716.

[65]   Remaker Deposition Tr. at 98:22-99:12 (Mar. 30, 2016).

[66]   *Id.* at 106:25-107:5.

[67]   *Id.* at 107:7-12.

110.   Accordingly, when the architects of Cisco's IOS[68] CLI decided to include a particular set of commands in the platform and to give the commands particular names and associated modes, they chose from a wide range of expressive options. And that is because a computer can be taught to understand and react to different words and multi-word inputs however the designer wants. In other words, the inventors of Cisco's IOS CLI commands did not have to include for technical reasons the specific words that are contained in the 500+ asserted command expressions. They were creative choices.

111.   By way of example, the command "show" is an expression of the idea or concept of displaying a particular configuration status of the device. There are many different ways to implement that idea and many different ways to even express that idea. For example, the word "display," "print," "watch," "view," or "info" are equally sufficient ways to express this idea. Other words such as "steve" or "book" or "phone" would be used just as well—a computer can recognize any combination of letters and numbers. Indeed, other vendors—such as Huawei— implement a command hierarchy using the command "display" instead of "show."

112.   Numerous Arista executives also have confirmed that the creation of command expressions is a creative process, that it is "very subjective," and that Arista is (and was) able to create new/different commands than the ones it copied from Cisco:

> 15 Q. Mr. Sweeney, based on your experience
> 16 working with CLI command syntax both at Cisco and at
> 17 Arista, would you agree that coming up with CLI
> 18 syntax for a particular command is very subjective?
> 19 A. It certainly is subjective, yes.
> 20 Q. And that means that different engineers
> 21 could come up with different ideas for the proper
> 22 CLI syntax for the same functionality, correct?
> 23 A. Yes.

---

[68]   Unless otherwise noted, my use of the term "IOS" refers to IOS, IOS-XR, IOS-XE, and NX-OS collectively.

6   Q.   So even with the same context it's
7 possible that you, as an engineer, may come to a
8 different professional judgment on the same question
9 today compared to how you would have seen the issue
10 back in 2003?
11   MR. FERRALL: Objection. Lacks
12 foundation. Vague and ambiguous.
13 BY MR. PAK:
14   Q.   Is that fair?
15   MR. FERRALL: Same objections.
16   THE WITNESS: You can always analyze
17 something and end up with a different conclusion.
18 In the CLI context I think you do learn over time
19 what the structure is, what are the other commands,
20 what is the language that everyone expects on that
21 device and then you adhere to that. There's
22 certainly learning there. And if you're trying to
23 debate, there are choices you can make. But you
24 need to know the full context to make the right

Anshul Sadana Deposition Tr. at 157:6-24.

113.   The copied command hierarchies also contain considerable original expression.

The decision to organize Cisco's IOS CLI commands into the designers' chosen hierarchy

reflects the original choices of the designers. As an illustration, a sub-command hierarchy for

"show" in Privileged EXEC mode is diagrammed below:

53
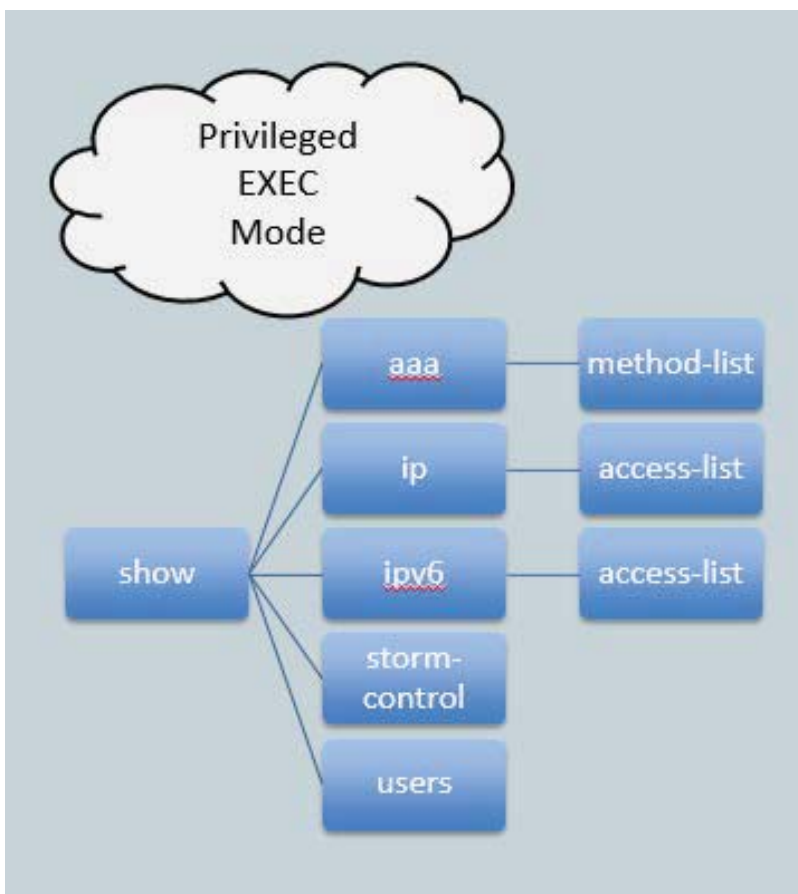
114.   Through this particular design, the designers were able to convey that a specific set of second words or tokens would follow the initial token, and then a further set, etc. The hierarchy conveys to a user an aesthetic sense of the set of choices, *i.e.,* what is possible and what is not. In some cases (*e.g.,* the use of "access-list" as an option under multiple higher level tokens), the hierarchy helps to organize choices into parallel possibilities.

115.   By branching initially on the dimension of "show" and then building out the hierarchy from there, the designers created an organizational structural that is aesthetically pleasing, easy to understand, and easier to remember (based on the subjective belief and professional judgment of Cisco's designers). A computer can execute the command "show_aaa_method-list" just as easily as it can execute a command called

"show_command_ipv6_access-list." The reason for choosing the organizational structure in the way that Cisco's designers did is so that they would have a unique command structure that Cisco's customers would easily be taught (again, based on the subjective belief and professional judgment of Cisco's designers) and because there was value in "the aesthetic of having a … symmetric hierarchy" that was "elegant."[69]

116. The decisions to organize Cisco's commands into modes with specific prompts reflects yet another conscious choice of expression. The command modes that I understand Cisco to be asserting in this case include "EXEC," "Privileged EXEC," "Global configuration," and "Interface configuration" (collectively, the "asserted command modes"). Rather than placing commands into different modes with unique prompts, the designers could have created a unified command structure without different modes and chosen a single prompt. Alternatively, Cisco's designers could have used different names for the asserted modes; for example, they could have chosen "ADMIN" instead of "EXEC" or "Secure ADMIN" instead of "Privileged EXEC." Similarly, "Universal setup" could have been chosen instead of "Global configuration" or "Edge setup" instead of "Interface configuration." Almost any other word choice could have been selected.

117. Further evidence that elements of Cisco's IOS CLI are creative is that they are what the user sees, what the user knows, and how the user talks to and interacts with the Cisco device. The user interface defines the user's experience. With the right selection of unique, intuitive commands and hierarchies—which Cisco endeavored to create on its own—Cisco built a successful business and became a market leader. That makes Cisco's IOS CLI distinctive compared to other competitors.

---

[69] Deposition Testimony of Kirk Lougheed Tr. at 128:10-129:19 (Nov. 20, 2015).

131.   The version information for the Cisco 4948E is as follows:

```
Switch>show version
Cisco IOS Software, Catalyst 4500 L3 Switch Software (cat4500e-IPBASE-M), Version
12.2(54)SG1, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2011 by Cisco Systems, Inc.
Compiled Thu 27-Jan-11 12:13 by prod_rel_team
Image text-base: 0x10000000, data-base: 0x12A709B8
ROM: 12.2(44r)SG12
Hobgoblin Revision 22, Fortooine Revision 1.22
Switch uptime is 1 minute
System returned to ROM by power-on
System image file is "bootflash:cat4500e-ipbase-mz.122-54.SG1.bin"
cisco WS-C4948E (MPC8548) processor (revision 5) with 1048576K bytes of memory.
Processor board ID CAT1552S66E
MPC8548 CPU at 1GHz, Cisco Catalyst 4948E
Last reset from PowerUp
1 Virtual Ethernet interface
48 Gigabit Ethernet interfaces
4 Ten Gigabit Ethernet interfaces
511K bytes of non-volatile configuration memory.
Configuration register is 0x2101
```

132.   The version information for the Arista 7010T is as follows:

```
localhost#show version
Arista DCS-7010T-48
Hardware version:    01.01
Serial number:       HSH16130550
System MAC address:  444c.a88f.f7f9

Software image version: 4.14.9.1M
Architecture:           i386
Internal build version: 4.14.9.1M-2714873.41491M
Internal build ID:      0c909198-32d7-4253-85d6-d0f013b47dbc

Uptime:              3 minutes
Total memory:        3907136 kB
Free memory:         1642992 kB
```

133.   The version information for the Arista 7554 is as follows:

```
localhost(s1)>show version
Arista DCS-7554-CH
Hardware version:    02.01
Serial number:       HSH14525015
System MAC address:  001c.7374.c093

Software image version: 4.14.5.1F-SSU
Architecture:           i386
Internal build version: 4.14.5.1F-SSU-2384023.EOS41451FSSU
Internal build ID:      11a6d19e-4978-481d-abfc-968034d5b2d1

Uptime:              1 minute
Total memory:        16012348 kB
Free memory:         13515668 kB
```

134.   The following independent forms of copying are covered by this report: (i) Arista's copying of copyrighted expressions in programs (including CLI commands, modes, hierarchies, prompts and screen outputs) from Cisco's copyrighted works into both physical and electronic media; (ii) Arista's copying of copyrighted expressions in documents from Cisco's copyrighted works into both physical and electronic media; and (iii) Arista's copying of copyrighted expressions in screen displays from Cisco's copyrighted works into both physical and electronic media. I understand that each one of these forms of copying are, alone, sufficient to establish copyright infringement.

A.   Arista Had Access To Cisco's Copyrighted Works & Admitted Copying

135.   Arista had access to Cisco's copyrighted works through a variety of sources, and, based on my inspection of the testimony and documents available in this case, it is my opinion that Arista has copied Cisco's copyrighted expressions in its IOS copyrighted works.

136.   Generally, Cisco's copyrighted documents such as its IOS-related manuals have been available to the public and on Cisco's website for years. Much of the Cisco documentation that I have personally observed contained a Cisco copyright notice, for example:

(IOS 11.0 (1989-1997), CSI-CLI-00430706)

Cisco ASR 1000 Series Aggregation Services Routers SIP and SPA Software Configuration Guide
© 2008 Cisco Systems, Inc. All rights reserved.

(IOS-XE 2.1 (2008), CSI-CLI-00229755)

Copyright © 2005 Cisco Systems, Inc. All rights reserved.

Text Part Number: OL-5903-04

(IOS-XR 3.2 (2005), CSI-CLl-00362851)

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

© 2008 Cisco Systems, Inc. All rights reserved.

(NX-OS 4.0 (2008), CSI-CLl-00362851)

137.   Cisco's products incorporating the IOS CLI copyrighted works have been and are publicly available as well (some well before the founding of Arista), and the Cisco operating systems that I inspected running on Cisco devices all have copyright notices on them. Therefore, anyone who sees IOS running or the related documentation is aware (or should be aware) that Cisco has legal rights associated with IOS and its related materials.

138.   More specifically, I understand that Arista gained access to Cisco's IOS copyrighted works by acquiring Cisco devices for purposes of performing analysis on those devices and did, in fact, analyze those Cisco devices, which contained the Cisco IOS copyrighted works, for both testing and development purposes. For example:[81]

> 20 Q. And did you ever -- during your time at
> 21 Arista when preparing competitive analysis, did you
> 22 ever look at any Cisco physical products?
> 23 A. I purchased, acquired Cisco hardware and
> 24 tested it in the lab.

---

[81]   Deposition of Sean Hafeez Tr. at 40:20-41:12.

Cisco Command Mode	Cisco Prompt	Identical or Similar Arista Command Mode	Identical or Similar Arista Prompt
User EXEC	router>	EXEC	switch>
Privileged EXEC	router#	Privileged EXEC	switch#
Global Configuration	router(config)#	Global Configuration	switch(config)#
Interface Configuration	router(config-if)#	Interface Configuration	switch(config-if)#
User EXEC	switch>	EXEC	switch>
Privileged EXEC	switch#	Privileged EXEC	switch#
EXEC	switch#	Privileged EXEC	switch#
Global Configuration	switch(config)#	Global Configuration	switch(config)#
Interface Configuration	switch(config-if)#	Interface Configuration	switch(config-if)#

183.   With respect to the modes, six out of eight modes are word-for-word identical. With respect to the three modes that are not verbatim copies, there is nevertheless 50% overlap between Arista's modes and Cisco's modes. The difference arises because Arista chose to use the mode "EXEC" instead of "User EXEC" (*i.e.*, dropping the single word "User") and "Privileged EXEC" instead of "EXEC" switches (*i.e.*, adding a single word "Privileged").

184.   With respect to the prompts, the switch-related prompts are identical across five prompts.   There are no differences. And when I compared Cisco's router prompts to Arista's switching prompts, I note that the overall structures are the same. The only difference is that Arista uses the term "switch" instead of "router".

185.   My analysis and conclusions apply to all versions of EOS accused in this case. My review of the materials in this case (including Arista's user manuals) shows that these modes and prompts have been used in every version of EOS accused in this case. *See* Exhibit Copying-4.

E.   **Cisco's CLI Command Hierarchy Compared to Arista's CLI Command Hierarchy**

186.   I understand that Cisco contends that Arista copied the following command

hierarchies as well:

- "aaa" command hierarchy
- "bgp" command hierarchy
- "clear" command hierarchy
- "dot1x" command hierarchy
- "ip" command hierarchy
- "ipv6" command hierarchy
- "neighbor" command hierarchy
- "show" command hierarchy
- "snmp-server" command hierarchy
- "spanning-tree" command hierarchy
- "vrrp" command hierarchy

187.   The command hierarchies in Amended Exhibit D1-D26 to Cisco's interrogatory

responses contain Cisco's copyrighted command expressions that Arista copied in whole or in

part by Arista. I have independently verified their contents and accuracy. I also have performed

testing on Cisco and Arista devices through which I confirmed the information in Amended

Exhibits D1-D26.

188.   The hierarchies contained in Amended Exhibits D1-D26 identify multi-word

command expressions. The hierarchies themselves are specific organizational and structural

elements of Cisco's copyrighted works.   The hierarchies are creative constructs that help a

user's thought process when interacting with a network device. The hierarchies can be expressed

visually by tree structures, for example, one "ip" hierarchy can be expressed as follows:
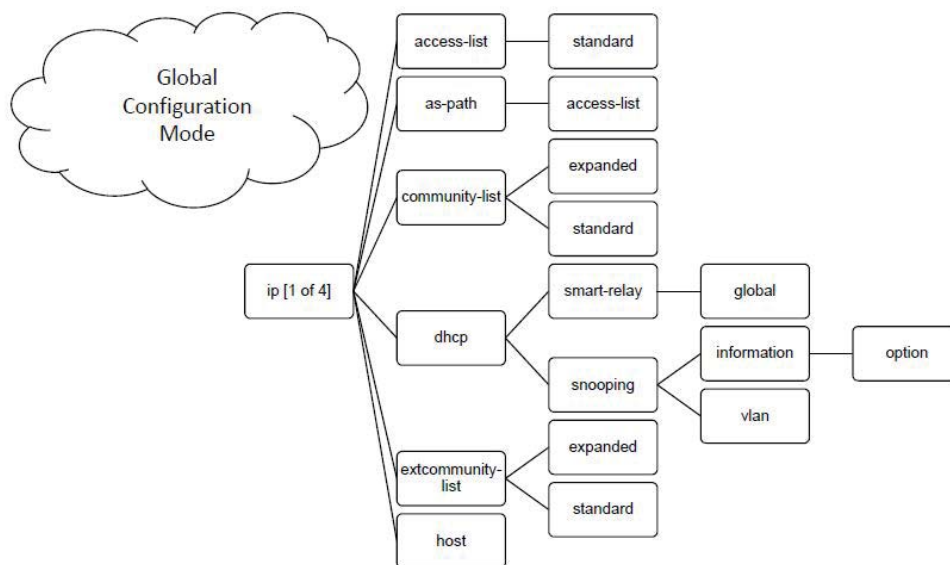
(D26 – NX-OS 6.2)

And a "show" hierarchy can be expressed as follows:
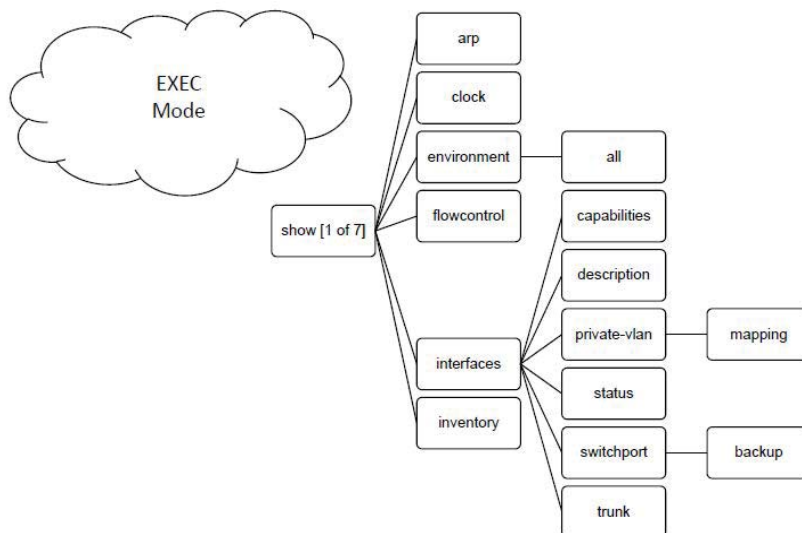
(D13 – IOS 15.4)

189.   Each command hierarchy is associated with a configuration mode (*e.g.*, the

"enable" EXEC command in EOS and IOS). What that means is that the command hierarchy

97

can only be used and accessed in that specific configuration mode. Associating modes with

hierarchies adds an additional layer to the organization and structure of the CLI.

190.   Based on my review of the documents, testimony, switches, and programs, it is

my opinion that Arista uses the same or similar multi-word commands, with the same or similar

multi-word syntaxes, organized into the same or similar hierarchies, and associated with the

same or similar modes that are in Cisco's copyrighted works.   These hierarchies are contained in

Cisco's and Arista's respective product documentation and programs.[164] In my opinion, Arista's

copying also is shown by its reproduction and use of the individual command expression in the

same modes as the commands are located in Cisco's copyrighted works.

191.   The following lists shows select examples of Arista's copying of hierarchies and

sub-hierarchies contained in Cisco's copyrighted works:[165]

"aaa" command hierarchy, including the following exemplary multiword command(s) within the "aa group server" sub-hierarchy:	"aaa group server radius" "aaa group server tacacs+"
"bgp" command hierarchy, including the following exemplary multiword command(s) within that hierarchy:	"bgp client-to-client reflection" "bgp confederation identifier"
"clear" command hierarchy, including the following exemplary multiword command(s) within the "clear ip" sub-hierarchy:	"clear ip igmp group" "clear ip nat translation"
"dot1x" command hierarchy, including the following exemplary multiword command(s)	"dot1x max-reauth-req"
"ip" command hierarchy, including the following exemplary multi-word command(s) within that hierarchy:	"ip as-path access-list"
"ip dhcp" sub-hierarchy, including the following exemplary multi-word command(s):	"ip dhcp snooping"
"ip igmp" sub-hierarchy, including the following exemplary multi-word command(s):	"ip igmp last-member-query-count"

[164]   CSI-CLI-00007473, CSI-CLI-00007244, CSI-CLI-00006858, CSI-CLI-00007841, CSI-CLI-00010517, CSI-CLI-00008985, CSI-CLI-00014141, CSI-CLI-00011973, CSI-CLI-00018146, CSI-CLI-00000084, CSI-CLI-00004616, CSI-CLI-00020575, CSI-CLI-00002332, CSI-CLI-00016001.

[165]   *See* Cisco's Second Amended Complaint.

	"ip igmp static-group"
"ip msdp" sub-hierarchy, including the following exemplary multi-word command(s):	"ip msdp sa-filter in"
"ip ospf" sub-hierarchy, including the following exemplary multi-word command(s):	"ip ospf shutdown" "ip ospf transmit-delay"
"ip pim" sub-hierarchy, including the following exemplary multi-word command(s):	"ip pim dr-priority" "ip pim query-interval"
"ipv6" command hierarchy, including "ipv6 nd" sub-hierarchy, including the following exemplary multi-word command(s):	"ipv6 nd managed-config-flag" "ipv6 nd ns-interval"
"ipv6 ospf" sub-hierarchy, including the following exemplary multi-word command(s):	"ipv6 ospf cost"
"neighbor" command hierarchy, including the following exemplary multi-word command(s):	"neighbor ebgp-multihop" "neighbor route-reflector-client"
"show" command hierarchy, including the following exemplary multiword command(s)[166]:	"show aaa method-lists"
"show interfaces" sub-hierarchy, including the following exemplary multi-word command(s):	"show interfaces private-vlan mapping" "show ip" sub-hierarchy (at least 50 matches), including:
"show ip bgp" sub-hierarchy (at least 8 matches), including the following exemplary multi-word command(s):	"show ip bgp regexp" "show ip mroute"
"show ipv6" sub-hierarchy, including "show ipv6 ospf" sub-hierarchy, including the following exemplary multi-word command(s):	"show ipv6 ospf border-routers" "show ipv6 route" sub-hierarchy
"snmp-server" command hierarchy (at least 12 matches), including the following exemplary multi-word command(s):	"snmp-server location"
"spanning-tree" command hierarchy, including the following exemplary multi-word command(s):	"spanning-tree bpduguard"
"vrrp" command hierarchy (at least 10 matches), including the following exemplary multiword command(s):	"vrrp timers advertise"

---

[166] Deposition of Sadana (Rough) Tr. at 98:17:17-23 (May 27, 2016) ("Show is a keyword for many of our commands. Q. And those would be the CLI commands that are used in EOS; correct? A. That's correct.").

99

192.   Additional examples that compare Cisco's command hierarchies illustrated visually in a tree structure compared to Arista's use of those same hierarchies in their documents are shown below:
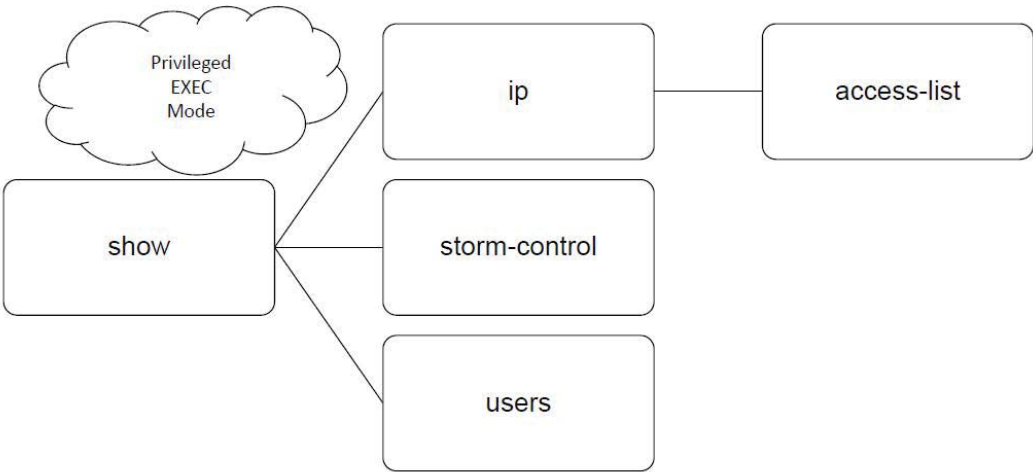
Example 1: "aaa" hierarchy
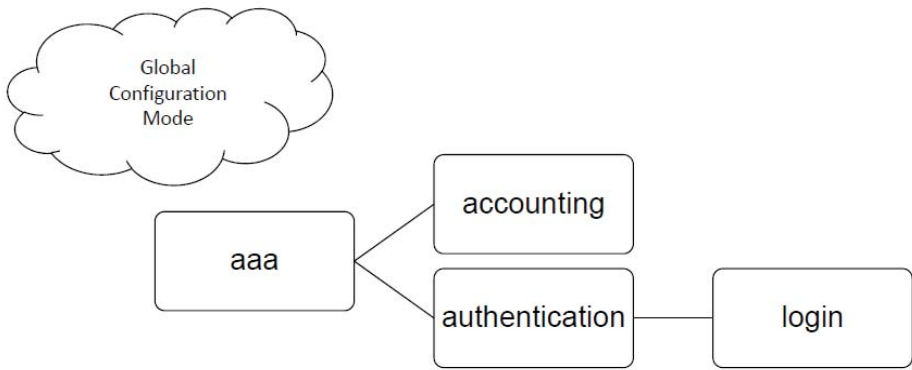
(D1, IOS Release 11.0)

(CSI-CLI-00007850, EOS 4.10.0)

Example 2: "show" hierarchy

(D1, IOS Release 11.0)

show ip access-lists 470
██████████████ 471
show storm-control 472

(CSI-CLl-00007850, EOS 4.10.0)

Example 3: another "show" hierarchy

(D1, IOS Release 11.0)

show privilege 132

101

(CSI-CLl-00007850, EOS 4.10.0)

Example 4: "bgp" hierarchy
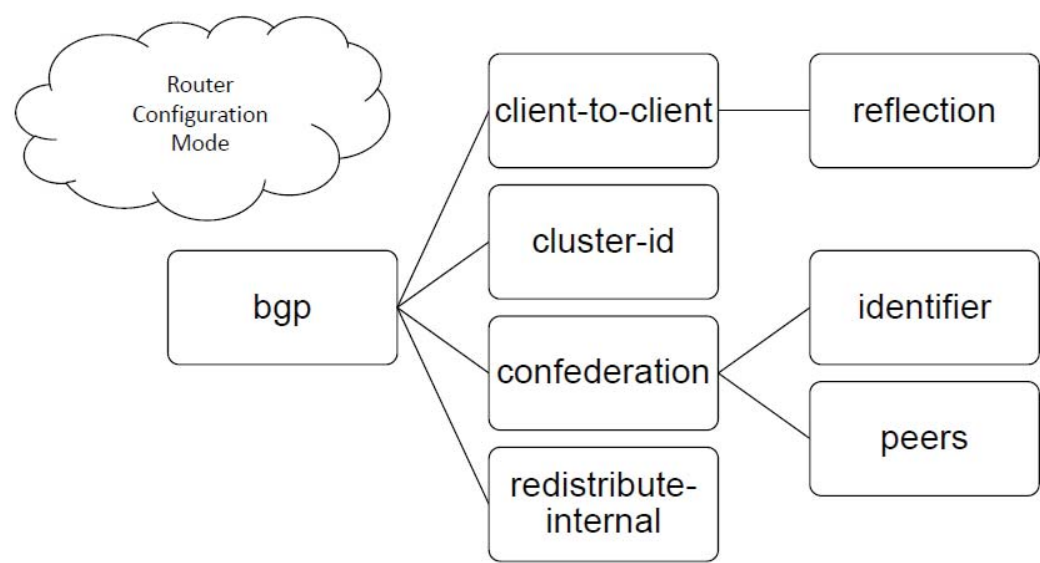
(D1, IOS Release 11.0)

(CSI-CLI-00018146, EOS 4.14.3F)

193.   Because the evidence of Arista's use of Cisco's command hierarchies is voluminous, I have summarized the similarities in Exhibit Copying-5, which is incorporated here by reference.

F.   Cisco's CLI Command Responses Compared to Arista's CLI Command Responses

194.   As explained above, another aspect of Cisco's CLI is the textual, screen output generated by the CLI as feedback when the user inputs a particular command. Cisco contends that in many instances, Arista provides output displays in EOS that are similar if not identical to the displays in Cisco's CLI. I agree that there are very close similarities between the screen outputs in Cisco's CLI and Arista's CLI. In some instance, in fact, it is almost impossible for a user to tell if they are using a Cisco device or an Arista device—the similarities are that close.

195.   To start, as I noted above in my summary of the direct evidence of Arista's copying, Arista's Vice President of Software Engineering specifically instructed Arista's employees to copy Cisco's CLI outputs:[167]

> 4 Q. Okay. Can you please read what you wrote
> 5 at the very top?
> 6 A. Sure. Let me just take a quick look at
> 7 what's here.
> 8 Q. Sure.
> 9 A. All right. What I wrote was:
> 10 "As I just replied on the support list,
> 11 I'm not sure that this request is
> 12 accurate. Unless IOS has changed
> 13 recently, I believe that we copy it
> 14 exactly on this one. I can't get to our
> 15 Cisco switch right now so we'll have to
> 16 double-check this when we can."
> ***
> 7 Q. Is it an output for a CLI command from

---

[167]   Deposition of Adam Sweeney Tr. at 317:4-318:9 (Jan. 29, 2016) (emphasis added); *see also* ARISTANDCA 10508650.

8 Cisco?

196.   9 A. Yes.

197.   This literal copying of text in the screen outputs is evident throughout Arista's

EOS. For example, Cisco's Exhibit E to its interrogatory response provides a listing of

command responses from Cisco's copyrighted works that Cisco's claims are identical or similar

in by Arista's EOS CLI. In addition, I understand that Cisco contends that Arista has copied the

non-literal elements of Cisco's command responses, including their structure, sequence and

organization as also shown in Exhibit E.

198.   I have independently confirmed that the examples shown in Exhibit E exist in

Arista's product documentation, and I agree that Arista's outputs are identical or similar to

Cisco's.

199.   For example, Cisco's CLI implements a command called "show snmp" which

displays the following output (show in Cisco's product documentation):[168]

---

[168]   CSI-CLI-00327842, Cisco IOS 15.4, Cisco IOS SNMP Support Command Reference
at 83, CSI-CLI-00327934 (2013).

104

```
Arista
------
localhost(s1)#show ip ospf ?
   border-routers          Border routers
   database                Database summary
   interface               Interface information
   lsa-log                 LSA throttling Log
   neighbor                Neighbor information
   request-list            Request list
   retransmission-list     Re-transmission list
   spf-log                 Spf Log
   vrf                     VRF name
   <1-65535>               Process ID
   >                       Redirect output to URL
   >>                      Append redirected output to URL
   |                       Output modifiers
   <cr>
```

(Arista)

```
Switch>show ip ospf ?
   <1-65535>               Process ID number
   border-routers          Border and Boundary Router Information
   database                Database summary
   interface               Interface information
   max-metric              Max-metric origination information
   mpls                    MPLS related information
   neighbor                Neighbor list
   sham-links              Sham link information
   statistics              Various OSPF Statistics
   summary-address         Summary-address redistribution Information
   timers                  OSPF timers information
   traffic                 Traffic related statistics
   virtual-links           Virtual link information
   |                       Output modifiers
   <cr>
```

(Cisco)

230.   Because the evidence of Arista's reproduction of Cisco's help descriptions into EOS is voluminous, I have summarized the similarities in Exhibit Copying-6, which is incorporated here by reference.

VII.   THERE IS NO INDUSTRY STANDARD FOR CISCO'S COPYRIGHTED WORKS

231.   I understand that Arista contends that it is permitted to use Cisco's IOS CLI because Cisco's IOS is an "industry standard." As explained below, I disagree with Arista and

124

have seen to actual evidence that Cisco's IOS CLI or its copyrighted works are part of any industry standard.

232.   When Cisco was founded in the mid-1980s, the router industry still was in its infancy; accordingly, there were no standards or constraints that required Cisco to choose a particular type of computer program for its products. There also was no clear consensus in the industry on which type of computer program was preferred. Thus, from the beginning of the design process, Cisco's engineers were faced with the fundamental creative question of what type of computer program to implement. And there were a broad range of options open to them, including but not limited to a command line interface (*e.g.*, DOS or UNIX), a graphical user interface, and a menu-driven interface, among others.

233.   I have discussed the first computer program—the command line interface—in great detail already and so I will not repeat an explanation of that interface here.

234.   A second type of computer program available to Cisco was a graphical user interface or "GUI." A GUI uses graphical symbols and icons to facilitate communication between the user and the router. Typically, a GUI includes windows, buttons and menus that can be accessed using a mouse. Each button and each option listed on the pull-down menus correspond to one or more functions that can be performed by the router. When a user clicks on a button or selects one of the menu options the router performs the functions associated with that button or menu option. Once of Cisco's competitors, Avaya, utilizes a graphical user interface for its routers,[175] for example:[176]

---

[175]   *See, e.g.*, https://www.avaya.com/usa/documents/avaya-ethernet-routing-switch-3500-series-lb7028.pdf ("For customers who are looking for a simple Graphical User Interface (GUI) for management and provisioning, Avaya's Enterprise Device Manager (EDM) is an embedded web-based element management and configuration tool that enables set-up, configuration and monitoring of a single device using either HTTP or HTTPS (Secure Web)");
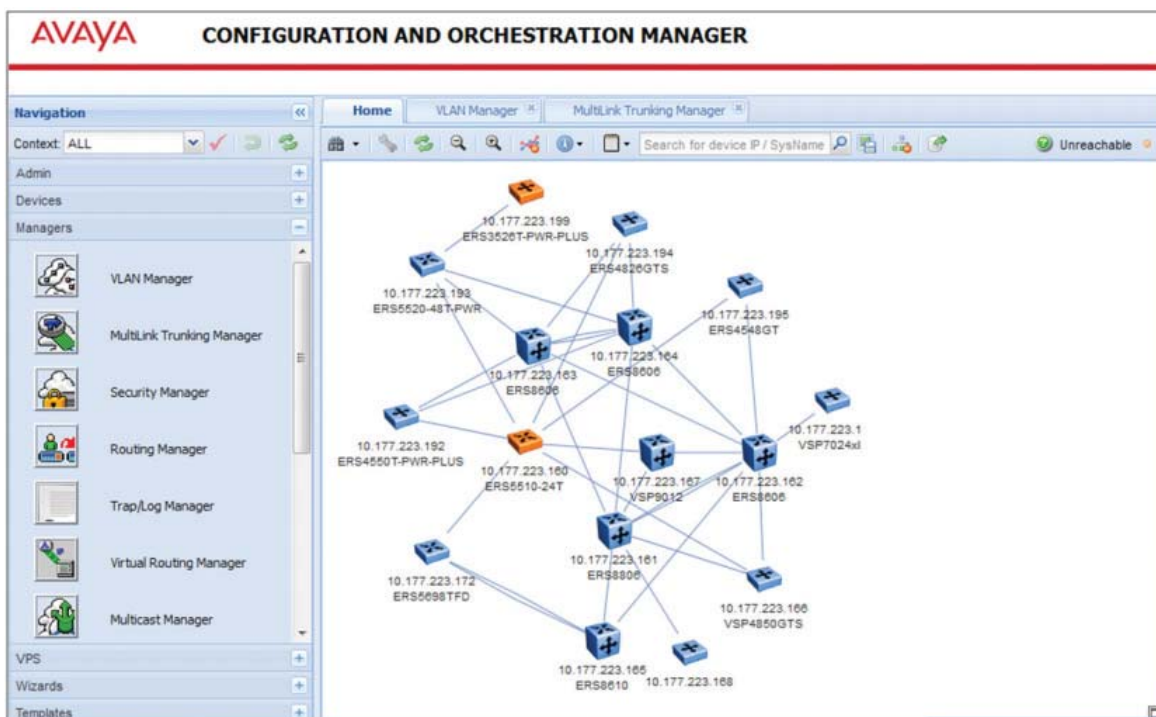
235.   A menu driven computer program presents the user with a series of menus, each of which lists several functions that can be performed by the router.   Unlike a graphical user interface, a menu-driven computer program is usually all text, with no graphical symbols or icons. When the user selections one of the options listed on the menu (by pressing a letter or number corresponding to the function) the router either performs the appropriate functions and/or presents the user with additional menus that permit the user to provide more detailed instructions.

236.   CLIs, GUIs, and menu-driven computer program all have been widely used since the original development of routers in the mid-1980s and remain in use today, by way of

---

see also https://www.avaya.com/usa/documents/avaya-ethernet-routing-switch-4000-series-dn4814.pdf.

176   *See* http://www.avaya.com/usa/product/configuration-and-orchestration-manager/.

example only (this is not meant to be an exhaustive list), Avaya has used a GUI,[177] 3Com has

used a menu-drive interface programs,[178] and Juniper (as discussed herein) as well as other

competitors in the market used CLI programs.

237.   Former and current Arista employees who worked on the development of Arista's

CLI have confirmed that different computer programs are available to customers.[179] For

example, Arista has a Linux offering for its switches that is an alternative to its default CLI:

> 4 Q. Okay. And then can you read the next—
> 5 actually, let's do the "In summary" statement here.
> 6 A. "In summary, it is possible to"—"to
> 7 be both a true modern network operating
> 8 system, and also run true Linux. The best
> 9 of both worlds is achievable with proper,
> 10 disciplined engineering."
> 11 Q. And is that a true statement --
> 12 A. I believe so.
> 13 Q.—from your perspective?
> 14 Okay. So what types of customers, based
> 15 on your experience, would approach the management
> 16 and configuration of Arista's switches using an
> 17 Arista Linux interface rather than the CLI interface
> 18 that we've been discussing?
> 19 A. I think customers are more likely to take
> 20 a UNIX oriented approach the more that they
> 21 internally are organized around—there's a notion
> 22 of Cloud computing in the industry. And Cloud
> 23 computing is the idea that you should build what's
> 24 called a Cloud, which is a uniform physical
> 25 infrastructure, including servers, storage, and
> 1 networking that supports a broad family of
> 2 applications, that you build the infrastructure once
> 3 and then allocate portions of the infrastructure to
> 4 specific applications on an and -- on a demand

---

[177]   https://downloads.avaya.com/css/P8/documents/100128482 (("The Configuration and Orchestration Manager (COM) is also a GUI")

[178]   http://h10032.www1.hp.com/ctg/Manual/c02608750 (the equivalent Web Interface menu").

[179]   Sean Hafeez Deposition Tr. at 67:8-11 ("Q. Now, you would agree with me, though, that there are different CLIs that are available to customers, right? A. Yes.").

5 basis.
6 This contrasts with the previous approach
7 where you build specific deployments of server
8 storage and networking targeted to a specific
9 application.
10 And within the Cloud world, a lot of Cloud
11 companies start with a UNIX kind of perspective
12 focusing on scaling servers, dealing with very large
13 numbers of servers at scale and so they develop
14 Linux expertise at managing those servers. And what
15 they find is that because of the way we've
16 integrated Linux into our switches they're able to
17 use those same kinds of technologies: Technologies
18 like Puppet and Chef, Ansible, CFEngine, to
19 administer their switches in a similar way to the
20 way they administer their servers.

Deposition Testimony of Kenneth Duda (Arista CTO & SVP of Software Engineering) at 145:4-

155:20 (Feb. 12, 2016).

14   Q. So it's true, sir, that rather than going
15 through the CLI interface an Arista customer could
16 use the dash Linux shell to configure some of the
17 settings of the Arista Networks switch, correct?
18   A. You could configure some things in Linux.
19 I don't think you could configure everything. It's
20 actually a little bit ironic, everything
23 well.
 ***
24   Q. Mr. Dale, my question is simple: Is --
25 it's possible that the Linux interface could be used
 1 by Arista customers to configure at least some of
 2 the settings in Arista's operating systems rather
 3 than using the CLI interface, correct?
 4   A. Yeah, I think within the extensible nature
 5 of EOS, the "E" in EOS, we enabled many ways that
 6 many customers could configure things. There were
 7 certainly some things that could be done in Linux.

its customers and distributors: Arista is a publicly traded company and derives value from the sale of products that contain computer programs and/or other works that infringe Cisco's copyrights. Furthermore, Arista has the right and ability to supervise at least the use, reproduction, distribution, and/or public display of computer programs and/or other works that infringe Cisco's copyrights by at least Arista's distributors and/or customers.

IX.   CISCO'S COPYRIGHTED WORKS WERE NOT COPIED FROM STANFORD

259.   I understand that Arista generally alleges that Cisco may not own its copyrighted works because they "are not Cisco's intellectual property, are derived from prior works over which Cisco has no ownership rights with respect to copyright assertions, and/or may not be asserted by Cisco in a copyright infringement action."[208] Specifically, I understand that Arista has made vague allegations relating to work Mr. Lougheed did at Stanford, on "TOPS-20," and worked related to SUMEX.[209]

260.   Although Arista has not formulated a clear theory or argument setting forth with any specificity which of the copyrighted works it contends came from Stanford or TOPS-20, I have nevertheless reviewed Mr. Lougheed's deposition testimony, spoken with Mr. Lougheed, and reviewed the source code relating to Arista's Stanford allegations.[210]   In sum, I have not seen any evidence that the multi-word command expressions (along with their specific associated modes and prompts) asserted in this case—or any of the other elements at issue in this case from the copyrighted works—originated from anywhere other than Cisco, nor have I seen any

---

[208]   Arista's response to Interrogatory No. 10.

[209]   *Id.*

[210]   *See also, e.g.,* KL-00000564; KL-00000186; KL-00000381; KL-00000655, KL-00000251; KL-SC-00000033 to 52; KL-00000001.

evidence to suggest that Cisco copied them. In fact, Cisco has provided voluminous information

detailing the provenance of the multi-word command expressions at issue in this case.[211]

261.   Further, in my opinion, the source code Mr. Lougheed worked on while at

Stanford is different than the source code he developed during that same time for Cisco. Mr.

Lougheed confirmed this to me as well.[212] The fact that certain single word commands or

protocols—e.g., "show," "clear," "help," "ip," "no," "arp," "bgp"—existed before Cisco does not

show (or prove) that any of Cisco's copyrighted works were copied, nor does it suggest to me

that the copyrighted works are unoriginal.   If Arista puts forth a more coherent and clear theory

or argument in its expert report that actually explains what its allegations are, I reserve the right

to supplement this report and/or respond to such allegations.

X.   CONCLUSION

262.   For presentation of my testimony at trial I may create and use demonstratives,

videos, and/or additional screenshots of the copyrighted works described in this report. In

addition, I may demonstrate the use of one or more Arista and Cisco switches at trial in support

of my testimony.

263.   I reserve the right to supplement or amend my opinions in response to opinions

expressed by Arista's experts, or in light of any additional evidence, testimony, discovery or

other information that may be provided to me after the date of this report. In addition, I reserve

the right to consider and testify about issues that may be raised by Arista's fact witnesses and

---

[211] *See* Cisco's responses to Arista's Interrogatory Nos. 2, 16, 19.

[212] Conversation with Kirk Lougheed (June 2, 2016); *see also* Lougheed Deposition Tr. 129:5-130:19, 166:24-169:16 ("I didn't like his lack of hierarchy"; "I started building a hierarchy") (Nov. 20, 2015); Lougheed Deposition Tr. 332:6-23, 339:18-340:9 (April 4, 2016).

experts at trial. I also reserve the right to modify or to supplement my opinions as a result of

ongoing expert discovery or testimony at trial.

      I certify under penalty of perjury that the foregoing is true and correct.

By: *Kevin C Almeroth*

Dr. Kevin C. Almeroth

June 3, 2016